



UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
-----------------	-------------	----------------------	---------------------	------------------

10/705,608

11/10/2003

Ravi L. Sahita

110466-153432

4694

31817 7590 10/25/2007
SCHWABE, WILLIAMSON & WYATT, P.C.
PACWEST CENTER, SUITE 1900
1211 S.W. FIFTH AVE.
PORTLAND, OR 97204

EXAMINER

ZHU, BO HUI ALVIN

ART UNIT

PAPER NUMBER

2619

MAIL DATE

DELIVERY MODE

10/25/2007

PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

Office Action Summary	Application No. 10/705,608	Applicant(s) SAHITA ET AL.	
	Examiner Bo Hui A. Zhu	Art Unit 2619	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

- 1) ☒ Responsive to communication(s) filed on 17 August 2007.
- 2a) ☒ This action is FINAL. 2b) ☐ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

- 4) ☒ Claim(s) 1-12,14-30,32-52-54-60 is/are pending in the application.
- 4a) Of the above claim(s) _____ is/are withdrawn from consideration.
- 5) ☐ Claim(s) _____ is/are allowed.
- 6) ☒ Claim(s) 1-12,14-30,32-52-54-60 is/are rejected.
- 7) ☐ Claim(s) _____ is/are objected to.
- 8) ☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on _____ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some * c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
 2. ☐ Certified copies of the priority documents have been received in Application No. _____.
 3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

- | | |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) | 4) <input type="checkbox"/> Interview Summary (PTO-413)
Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948) | 5) <input type="checkbox"/> Notice of Informal Patent Application |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO/SB/08)
Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____ |

DETAILED ACTION

Response to Amendment

1. The amendment filed on August 17, 2007 has been entered.
Claims 1 – 12, 14 – 30, 32 – 52 and 54 - 60 are pending.
Claims 1 – 12, 14 – 30, 32 – 52 and 54 - 60 are rejected.
The 112 2nd paragraph rejections of claims 16 – 18 and 56 - 58 have been withdrawn in view of the amendment to the claims.

Claim Rejections - 35 USC § 101

2. 35 U.S.C. 101 reads as follows:

Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor, subject to the conditions and requirements of this title.
3. Claims 41 – 60 are rejected under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter.

Regarding claims 41 – 60, the subject matter of having a plurality of machine accessible programming instructions stored on a storage medium and configured to program a computing device is non-statutory. It is recommended by the Examiner that the claim language be changed to recite a computer-readable medium encoded or embodied with a computer program instructions, so the claim is drawn to a computer element which, when executed by a computer, defines structural and functional interrelationships between the instructions and the computer to permit the instructions functionality to be realized, and is thus statutory. Please see pages 30 and 53 of the

Art Unit: 2619

Interim Guidelines for Examination of Patent Applications for Patent Subject Matter

Eligibility

Claim Rejections - 35 USC § 112

4. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

5. Claims 5, 6, 21 – 30, 32 – 40, 45 and 46 are rejected under 35 U.S.C. 112, second paragraph, as being indefinite for failing to particularly point out and distinctly claim the subject matter which applicant regards as the invention.

(1) with regard to claims 5, 6, 45 and 46 :

Claims 5 and 45 recites selecting a rule from a Rules Table. Claims 2 and 42, the respective parent claims of claims 5 and 45, recite parsing a cached Rules Table. It is not clear whether the two Rules Tables are referring to the same Rules Table or otherwise. If the two Rules Tables are referring to the same Rules Table, it is recommended by the Examiner that the second recitation of Rules Table be changed to “the Rules Table” instead of “a Rules Table” for clarity of subject matter.

(2) with regard to claims 21 – 30 and 32 – 40:

In claim 21, the claim limitation "the Condition Set Tables", "the Rule Group's field", "the first Condition Set", "the rule", "the pointers" and "the cache memory" lack antecedence basis. And it is not clear what it means by pointers being grouped by Condition Set. Condition Set lacks antecedence basis in the claim.

Claim Rejections - 35 USC § 103

6. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

7. Claims 1 – 9, 11 and 16 – 20 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703).

(1) with regard to claim 1:

Albert et al. discloses a system and method, comprising: receiving a packet (1100 on Fig. 11); applying an Active Rule to the received Packet (1110 on Fig. 11; the wildcard affinity is considered as an Active Rule); accessing a cached Condition Set Table having at least one Condition Set, associated with the Active Rule; for each Condition Set, having at least one Condition, in the Condition Set Table, evaluating the Condition(s) in the Condition Set; determining if the Condition Set is met (paragraph [0232] – [0233]; in Fig. 12, 1202, 1204, 1206, 1208 and 1210 are the conditions and

together they construct a condition set, a condition set table); determining if the Active Rule is met (wildcard affinity found or not found on Fig. 11); and executing an Action Set associated with the Active Rule (1114 on Fig. 11; the action associated with the affinity key is the action set of the affinity).

Albert et al. however does not disclose utilizing the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set; and utilizing the pointers to access a Condition Table having the Condition.

Relan teaches utilizing the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set and utilizing the pointers to access a Condition Table having the Condition (Fig. 6; paragraph [0069]).

It would have been desirable to utilize the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set and utilize the pointers to access a Condition Table having the Condition, because it would improve the flexibility and efficiency of the system and using pointers as references to another object would greatly reduce the complexity in referencing different objects in data structure of a system. Therefore, it would have been desirable to one of ordinary skill in the art at the time of the invention to include the method as taught by Relan into the system of Albert et al.

(2) with regard to claim 2:

Albert et al. further discloses that parsing a cached Rules Tables, having a plurality of rules, to determine if a rule is pertinent to the received packet; if so, making the pertinent rule the Active Rule (paragraphs [0102], [0119] – [0126], a affinity is a rule and the affinity key of the affinity determines if the rule is pertinent to a packet or not).

(3) with regard to claim 3:

Albert et al. further discloses that if more than one rule is pertinent, performing the method of claim 1 for each pertinent rule (paragraph [0102] teaches two actions could have taken place for a packet that matches the affinity key).

(4) with regard to claim 4:

Albert et al. further discloses that the rules includes a source, a destination, and a protocol; and determining if a rule is pertinent to the received packet includes: determining if the source of the received packet and the source of the rule are equivalent; determining if the destination of the received packet and the destination of the rule are equivalent; determining if the protocol of the received packet and the protocol of the rule are equivalent; if all three are equivalent, considering the rule pertinent to the received packet (Fig. 12).

(5) with regard to claim 5:

Albert et al. further disclose selecting a rule; and accessing a Rule Group from a Rules Group Table (Fig. 12, 1202 – 1210 constructs a Rule Group) wherein the Rule Group includes a field to facilitate access to the first Condition Set associated with the rule, and a field to facilitate access to the first Action Set associated with the rule (each part of the Rule Group, i.e. 1202, 1204 ... 1210, is considered as a field).

(6) with regard to claim 6:

Albert et al. further discloses that accessing the Condition Set Tables utilizing the Rule Group's field to facilitate access to the first Condition Set associated with the rule (each field, namely 1202 – 1210, is utilized to evaluate if there is a match with the rule or not).

(7) with regard to claim 7:

Albert et al. further discloses that each of the Conditions includes pattern and an opcode; and for each Condition, comparing the pattern to the received packet in the manner dictated by the opcode, and producing a Boolean value as a result of the comparison (paragraph [0232]; for example, in reference to Fig. 12, for condition 1202, the pattern is the protocol of the affinity and the opcode is equal or the same and the the Boolean result is either YES or NO); an wherein determining if the Condition Set is met includes: computing a single Boolean value utilizing the Boolean values resulting from evaluating the Condition(s) (the single Boolean result is indicated by 1211 and 1212, i.e. either a MATCH or NO MATCH).

(8) with regard to claim 8:

Albert et al. further discloses that each of the at least one Conditions further includes a pattern mask to alter interpretation of the pattern (1204, affinity mask; affinity mask alters the interpretation of the affinity source address).

(9) with regard to claim 9:

Albert et al. further discloses that each of the at least one Conditions further includes a flag to denote that the Condition has already been evaluated for the current

Art Unit: 2619

received packet, and a value denoting the result of that evaluation (Fig. 12, the result YES or NO of each condition is a flag to denote that the condition has been evaluated and result of that evaluation).

(10) with regard to claim 11:

Albert et al. further discloses computing a single Boolean value utilizing the Boolean values resulting from determining if the Condition Set is met (an affinity is determined to be either found or not found, based on the result of the comparison of rules associated with it).

(11) with regard to claim 14:

Albert et al. does not disclose any Condition is included by a plurality of Condition Sets.

Relan discloses any Condition is included by a plurality of Condition Set (conditional expression entries are viewed as the plurality of condition sets).

It would have been desirable to have a plurality of condition sets to include any condition because it would improve the flexibility and efficiency of the system as the system could react to more condition sets. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include the method as taught by Relan in to the system of Albert et al.

(12) with regard to claim 16:

Albert et al. further discloses executing an Action Set associated with the Active Rule includes: accessing an Action Set having at least one Action; and executing each

Art Unit: 2619

Action within the Action Set (1104 on Fig. 11; the action associated with the affinity key is the action set of the affinity).

(13) with regard to claim 17:

Albert et al. further discloses that executing each Action includes performing reporting information to a third party (paragraph [0145]).

(14) with regard to claim 18:

Albert et al. further discloses that accessing a Rule Group having a pointer to the Action Set (Fig. 12, 1202 – 1210 constructs a rule group, and the result of this rule group follows by the action associated with the affinity matches the rules); accessing an Action Set Table having a plurality of Action Sets and selecting an Action Set from the Action Set Table (actions [0145] – [0152] constructs a action set table; and the action associated with an affinity is a specific action selected from this set of action).

(15) with regard to claims 19 and 20:

Albert et al. further discloses that the number of Conditions in a Condition Set and the number of Actions in an Action Set is limited, at least in part, by the amount of information that can be read from a cache memory in one clock cycle (the number of conditions and actions that can be read from a memory is inherently limited by the speed of the memory).

8. Claims 10 and 12 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and further in view of Shimokawa et al. (US 4,298,933).

(1) with regard to claim 10:

Albert et al. further discloses logically AND, as each Condition's Boolean value is computed, the Boolean values resulting from evaluating the Condition(s) (Fig. 12 and paragraph [0232] teaches that the result of each condition are logically ANDed, i.e. all have to be YES in order to a MATCH, either one is NO will result in a NO MATCH). And since Albert et al. discloses the result of each condition comparison only requires 1 bit (to represent 1 or 0 representing YES or NO), a 1-bit device is used for performing this method.

Albert et al. however does not expressly disclose utilizing a Condition Accumulator to perform such method.

Shimokawa et al. teaches the use of accumulator for storing intermediate arithmetic and logic results (Bit Accumulator, 32 on Fig. Fig. 5). It would have been desirable to use an accumulator for storing intermediate logic results because it would eliminate the need for writing the result of each calculation to a separate memory device, thus improve the efficiency of the operation. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use accumulator for storing logical operation results in the system of Albert et al.

(2) with regard to claim 12:

Albert et al. further discloses logically OR, as each Condition Set's Boolean value is computed, the Boolean values resulting from determining if the Condition Set is met (Fig. 12 and paragraph [0232] teaches that the final result is either a MATCH or NO MATCH, this indicates that MATCH or NO MATCH is logically ORed). And since Albert et al. discloses the final result only requires 1 bit (to represent 1 and 0 representing a MATCH or NO MATCH), a 1-bit device is used for performing this method.

Albert et al. however does not expressly disclose utilizing a Condition Accumulator to perform that method.

Shimokawa et al. teaches the use of accumulator for storing intermediate arithmetic and logic results (Bit Accumulator, 32 on Fig. Fig. 5). It would have been desirable to use an accumulator for storing intermediate logic results because it would eliminate the need for writing the result of each calculation to a separate memory device, thus improve the efficiency of the operation. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use accumulator for storing logical operation results in the system of Albert et al.

9. Claim 15 is rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and further in view of Egbert et al. (US 6,115,387).

(1) with regard to claim 15:

Albert et al. does not disclose the Condition Indirection Table is stored within a Content Addressable Memory (CAM).

Egbert et al. teaches using a content addressable memory to store a table that stores pointers (161, on Fig. 14).

It would have been desirable to use a content addressable memory to store tables because it would reduce system latency as table look-ups with content address memory is very fast and efficient. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include the method as taught by Egbert et al. in to the system of Albert et al.

10. Claims 21 – 27, 29 and 33 – 38 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and further in view of Kametani (US 7,099,324).

(1) with regard to claim 21:

Albert et al. discloses an apparatus, comprising: a micro-engine having a rule based action packet processing engine that is capable of processing a received packet (Fig. 11; paragraphs [0226] – [0228]); a network processor core that is capable of resource management and control of the micro-engine (252 on Fig. 2B); a packet buffer to receiver a packet (260 on Fig. 2B; [0084]); and a cache memory to store data structures for the micro-engine (254 on Fig. 2B).

Albert et al. however does not disclose accessing the Condition Set Table utilizing the Rule Group's field to facilitate access to the first Condition Set associated with the rule, utilizing the Condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set; and utilizing the pointers to access a Condition Table having the Condition, and wherein the Condition Set Table is stored as a data structure within the cache memory.

Relan teaches accessing the Condition Set Table utilizing the Rule Group's field to facilitate access to the first Condition Set associated with the rule, utilizing the Condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set; and utilizing the pointers to access a Condition Table having the Condition (Fig. 6; paragraph [0069]).

It would have been desirable to utilize the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set and utilize the pointers to access a Condition Table having the Condition, because it would improve the flexibility and efficiency of the system and using pointers as references to another object would greatly reduce the complexity in referencing different objects in data structure of a system. Therefore, it would have been desirable to one of ordinary skill in the art at the time of the invention to include the method as taught by Relan into the system of Albert et al.

Kametani teaches storing a table in the cache memory (17 and 17a, on Fig. 1).

It would have been desirable to store tables as data structure in the cache memory because it would expedite the process of accessing the tables as data

scored in the cache memory are ready to be processed by system processors.

Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include the method as taught by Kametani in to the system of Albert et al.

(2) with regard to claim 22:

Albert et al. discloses all of the subject matter as discussed above but fails to disclose that a plurality of received packets can be processed substantial simultaneously.

The Examiner takes Official Notice that the method of parallel processing is well known in the art. Using a plurality of the same processing unit can process a plurality of packet simultaneously. It would be desirable because it would improve the efficiency of the system by processing multiple packets at the same time. Therefore, it would have been obvious to one of ordinary skill in the art to implement parallel processing in system of Albert et al. in order to improve system efficiency.

(3) with regard to claim 23:

Albert et al. discloses receiving a packet (1100 on Fig. 11); applying an Active Rule to the received Packet (1110 on Fig. 11; the wildcard affinity is considered as an Active Rule); accessing a cached Condition Set Table having at least one Condition Set, associated with the Active Rule; for each Condition Set, having at least one Condition, in the Condition Set Table, evaluating the Condition(s) in the Condition Set; determining if the Condition Set is met (paragraph [0232] – [0233]; in Fig. 12, 1202, 1204, 1206, 1208 and 1210 are the conditions and together they construct a condition set, a condition set table); determining if the Active Rule is met (wildcard affinity found

or not found on Fig. 11); and executing an Action Set associated with the Active Rule (1114 on Fig. 11; the action associated with the affinity key is the action set of the affinity).

(4) with regard to claim 24:

Albert et al. further discloses that parsing a cached Rules Tables, having a plurality of rules, to determine if a rule is pertinent to the received packet; if so, making the pertinent rule the Active Rule (paragraphs [0102], [0119] – [0126], a affinity is a rule and the affinity key of the affinity determines if the rule is pertinent to a packet or not).

(5) with regard to claim 25:

Albert et al. further discloses that the rules includes a source, a destination, and a protocol; and determining if a rule is pertinent to the received packet includes: determining if the source of the received packet and the source of the rule are equivalent; determining if the destination of the received packet and the destination of the rule are equivalent; determining if the protocol of the received packet and the protocol of the rule are equivalent; if all three are equivalent, considering the rule pertinent to the received packet (Fig. 12).

(6) with regard to claim 26:

Albert et al. further disclose selecting a rule; and accessing a Rule Group from a Rules Group Table (Fig. 12, 1202 – 1210 constructs a Rule Group) wherein the Rule Group includes a field to facilitate access to the first Condition Set associated with the rule, and a field to facilitate access to the first Action Set associated with the rule (each part of the Rule Group, i.e. 1202, 1204 ... 1210, is considered as a field).

(7) with regard to claim 27:

Albert et al. further discloses that each of the Conditions includes pattern and an opcode; and for each Condition, comparing the pattern to the received packet in the manner dictated by the opcode, and producing a Boolean value as a result of the comparison (paragraph [0232]; for example, in reference to Fig. 12, for condition 1202, the pattern is the protocol of the affinity and the opcode is equal or the same and the Boolean result is either YES or NO); and wherein determining if the Condition Set is met includes: computing a single Boolean value utilizing the Boolean values resulting from evaluating the Condition(s) (the single Boolean result is indicated by 1211 and 1212, i.e. either a MATCH or NO MATCH).

(8) with regard to claim 29:

Albert et al. further discloses computing a single Boolean value utilizing the Boolean values resulting from determining if the Condition Set is met (an affinity is determined to be either found or not found, based on the result of the comparison of rules associated with it).

(9) with regard to claim 33:

Albert et al. further discloses executing an Action Set associated with the Active Rule includes: accessing an Action Set having at least one Action; and executing each Action within the Action Set (1104 on Fig. 11; the action associated with the affinity key is the action set of the affinity).

(10) with regard to claim 34:

Albert et al. further discloses that executing each Action includes performing reporting information to a third party (paragraph [0145]).

(11) with regard to claim 35:

Albert et al. further discloses that accessing a Rule Group having a pointer to the Action Set (Fig. 12, 1202 – 1210 constructs a rule group, and the result of this rule group follows by the action associated with the affinity matches the rules); accessing an Action Set Table having a plurality of Action Sets and selecting an Action Set from the Action Set Table (actions [0145] – [0152] constructs a action set table; and the action associated with an affinity is a specific action selected from this set of action).

(12) with regard to claims 36 and 37:

Albert et al. further discloses that the number of Conditions in a Condition Set and the number of Actions in an Action Set is limited, at least in part, by the amount of information that can be read from a cache memory in one clock cycle (the number of conditions and actions that can be read from a memory is inherently limited by the speed of the memory).

(13) with regard to claim 38:

Albert et al. discloses all of the subject matter as discussed above but fails to disclose the cache memory includes a SRAM.

The Examiner takes Official Notice that use of SRAM (Static Random Access Memory) in memory is well known in the art. It would be desirable to use SRAM because it is faster and less power consumed than DRAM. Thus it would have been

obvious to one of ordinary skill in the art at the time of the invention to use SRAM in the system of Albert et al.

11. Claim 32 is rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and Kametani (US 7,099,324) and further in view of Egbert et al. (US 6,115,387).

(1) with regard to claim 32:

Albert et al. does not disclose the Condition Indirection Table is stored within a Content Addressable Memory (CAM).

Egbert et al. teaches using a content addressable memory to store a table that stores pointers (161, on Fig. 14).

It would have been desirable to use a content addressable memory to store tables because it would reduce system latency as table look-ups with content address memory is very fast and efficient. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include the method as taught by Egbert et al. in to the system of Albert et al.

12. Claims 28 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of in view of Relan (US 2003/0174703) and Kametani (US 7,099,324) and further in view of Shimokawa et al. (US 4,298,933).

(1) with regard to claim 28:

Albert et al. further discloses logically AND, as each Condition's Boolean value is computed, the Boolean values resulting from evaluating the Condition(s) (Fig. 12 and paragraph [0232] teaches that the result of each condition are logically ANDed, i.e. all have to be YES in order to a MATCH, either one is NO will result in a NO MATCH). And since Albert et al. discloses the result of each condition comparison only requires 1 bit (to represent 1 or 0 representing YES or NO), a 1-bit device is used for performing this method.

Albert et al. however does not expressly disclose utilizing a Condition Accumulator to perform such method.

Shimokawa et al. teaches the use of accumulator for storing intermediate arithmetic and logic results (Bit Accumulator, 32 on Fig. Fig. 5). It would have been desirable to use an accumulator for storing intermediate logic results because it would eliminate the need for writing the result of each calculation to a separate memory device, thus improve the efficiency of the operation. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use accumulator for storing logical operation results in the system of Albert et al.

(2) with regard to claim 30:

Albert et al. further discloses logically OR, as each Condition Set's Boolean value is computed, the Boolean values resulting from determining if the Condition Set is met (Fig. 12 and paragraph [0232] teaches that the final result is either a MATCH or NO MATCH, this indicates that MATCH or NO MATCH is logically ORed). And since Albert

Art Unit: 2619

et al. discloses the final result only requires 1 bit (to represent 1 and 0 representing a MATCH or NO MATCH), a 1-bit device is used for performing this method.

Albert et al. however does not expressly disclose utilizing a Condition Accumulator to perform that method.

Shimokawa et al. teaches the use of accumulator for storing intermediate arithmetic and logic results (Bit Accumulator, 32 on Fig. Fig. 5). It would have been desirable to use an accumulator for storing intermediate logic results because it would eliminate the need for writing the result of each calculation to a separate memory device, thus improve the efficiency of the operation. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use accumulator for storing logical operation results in the system of Albert et al.

13. Claims 39 and 40 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and Kametani (US 7,099,324) and further in view of Harriman, Jr. (US 5,398,245).

(1) with regard to claim 39:

Albert et al. discloses all of the subject matter as discussed above but fails to disclose the packet buffer include a DRAM.

Harriman, Jr. teaches using DRAM to buffer packet received (column 3, lines 51 – 53).

It would be desirable to use DRAM as packet buffer because DRAM is less expensive than SRAM. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use DRAM in the system of Albert et al.

(2) with regard to claim 40:

Albert et al. further discloses the network processor core is capable of receiving instructions via a generic programmable interface (258 on Fig. 2B); and the received instructions are capable of altering the Condition Set and the Action Set (paragraph [0169]).

14. Claims 41 – 49, 51 and 56 - 60 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and further in view of Langberg et al. (US 5,852,630).

(1) with regard to claim 41:

Albert et al. discloses a system and method, comprising: receiving a packet (1100 on Fig. 11); applying an Active Rule to the received Packet (1110 on Fig. 11; the wildcard affinity is considered as an Active Rule); accessing a cached Condition Set Table having at least one Condition Set, associated with the Active Rule; for each Condition Set, having at least one Condition, in the Condition Set Table, evaluating the Condition(s) in the Condition Set; determining if the Condition Set is met (paragraph [0232] – [0233]; in Fig. 12, 1202, 1204, 1206, 1208 and 1210 are the conditions and together they construct a condition set, a condition set table); determining if the Active

Rule is met (wildcard affinity found or not found on Fig. 11); and executing an Action Set associated with the Active Rule (1114 on Fig. 11; the action associated with the affinity key is the action set of the affinity).

Albert et al. however does not disclose utilizing the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set; and utilizing the pointers to access a Condition Table having the Condition.

Relan teaches utilizing the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set and utilizing the pointers to access a Condition Table having the Condition (Fig. 6; paragraph [0069]).

It would have been desirable to utilize the condition Set Table to access a Condition Indirection Table having a pointer to each Condition wherein the pointers are grouped by Condition Set and utilize the pointers to access a Condition Table having the Condition, because it would improve the flexibility and efficiency of the system and using pointers as references to another object would greatly reduce the complexity in referencing different objects in data structure of a system. Therefore, it would have been desirable to one of ordinary skill in the art at the time of the invention to include the method as taught by Relan into the system of Albert et al.

Albert et al. does not disclose using a computer readable medium with instruction codes to perform the method as discussed above.

Langberg et al. teaches a method for a transceiver warm start activation procedure can be implemented in software stored in a computer-readable medium. The computer-readable medium is an electronic, magnetic, optical, or other physical device or means that can contain or store a computer program for use by or in connection with a computer-related system or method (column 3, lines 51-65). Using a computer readable medium with program instruction code would be desirable because it would perform the same function of using hardware but offer the advantage of less expense, adaptability and flexibility. Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to include the limitation as taught by Langberg et al. into the system of Albert et al. so as to reduce cost and improve the adaptability and flexibility of the logic simulation.

(2) with regard to claim 42:

Albert et al. further discloses that parsing a cached Rules Tables, having a plurality of rules, to determine if a rule is pertinent to the received packet; if so, making the pertinent rule the Active Rule (paragraphs [0102], [0119] – [0126], a affinity is a rule and the affinity key of the affinity determines if the rule is pertinent to a packet or not).

(3) with regard to claim 43:

Albert et al. further discloses that if more than one rule is pertinent, performing the method of claim 1 for each pertinent rule (paragraph [0102] teaches two actions could have taken place for a packet that matches the affinity key).

(4) with regard to claim 44:

Albert et al. further discloses that the rules includes a source, a destination, and a protocol; and determining if a rule is pertinent to the received packet includes: determining if the source of the received packet and the source of the rule are equivalent; determining if the destination of the received packet and the destination of the rule are equivalent; determining if the protocol of the received packet and the protocol of the rule are equivalent; if all three are equivalent, considering the rule pertinent to the received packet (Fig. 12).

(5) with regard to claim 45:

Albert et al. further disclose selecting a rule; and accessing a Rule Group from a Rules Group Table (Fig. 12, 1202 – 1210 constructs a Rule Group) wherein the Rule Group includes a field to facilitate access to the first Condition Set associated with the rule, and a field to facilitate access to the first Action Set associated with the rule (each part of the Rule Group, i.e. 1202, 1204 ... 1210, is considered as a field).

(6) with regard to claim 46:

Albert et al. further discloses that accessing the Condition Set Tables utilizing the Rule Group's field to facilitate access to the first Condition Set associated with the rule (each field, namely 1202 – 1210, is utilized to evaluate if there is a match with the rule or not).

(7) with regard to claim 47:

Albert et al. further discloses that each of the Conditions includes pattern and an opcode; and for each Condition, comparing the pattern to the received packet in the manner dictated by the opcode, and producing a Boolean value as a result of the

Art Unit: 2619

comparison (paragraph [0232]; for example, in reference to Fig. 12, for condition 1202, the pattern is the protocol of the affinity and the opcode is equal or the same and the the Boolean result is either YES or NO); an wherein determining if the Condition Set is met includes: computing a single Boolean value utilizing the Boolean values resulting from evaluating the Condition(s) (the single Boolean result is indicated by 1211 and 1212, i.e. either a MATCH or NO MATCH).

(8) with regard to claim 48:

Albert et al. further discloses that each of the at least one Conditions further includes a pattern mask to alter interpretation of the pattern (1204, affinity mask; affinity mask alters the interpretation of the affinity source address).

(9) with regard to claim 49:

Albert et al. further discloses that each of the at least one Conditions further includes a flag to denote that the Condition has already been evaluated for the current received packet, and a value denoting the result of that evaluation. (Fig. 12, the result YES or NO of each condition is a flag to denote that the condition has been evaluated and result of that evaluation).

(10) with regard to claim 51:

Albert et al. further discloses computing a single Boolean value utilizing the Boolean values resulting from determining if the Condition Set is met (an affinity is determined to be either found or not found, based on the result of the comparison of rules associated with it).

(11) with regard to claim 54:

Albert et al. does not disclose any Condition is included by a plurality of Condition Sets.

Relan discloses any Condition is included by a plurality of Condition Set (conditional expression entries are viewed as the plurality of condition sets).

It would have been desirable to have a plurality of condition sets to include any condition because it would improve the flexibility and efficiency of the system as the system could react to more condition sets. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include the method as taught by Relan in to the system of Albert et al.

(12) with regard to claim 56:

Albert et al. further discloses executing an Action Set associated with the Active Rule includes: accessing an Action Set having at least one Action; and executing each Action within the Action Set (1104 on Fig. 11; the action associated with the affinity key is the action set of the affinity).

(13) with regard to claim 57:

Albert et al. further discloses that executing each Action includes performing reporting information to a third party (paragraph [0145]).

(14) with regard to claim 58:

Albert et al. further discloses that accessing a Rule Group having a pointer to the Action Set (Fig. 12, 1202 – 1210 constructs a rule group, and the result of this rule group follows by the action associated with the affinity matches the rules); accessing an Action Set Table having a plurality of Action Sets and selecting an Action Set from the

Action Set Table (actions [0145] – [0152] constructs a action set table; and the action associated with an affinity is a specific action selected from this set of action).

(15) with regard to claims 59 and 60:

Albert et al. further discloses that the number of Conditions in a Condition Set and the number of Actions in an Action Set is limited, at least in part, by the amount of information that can be read from a cache memory in one clock cycle (the number of conditions and actions that can be read from a memory is inherently limited by the speed of the memory).

15. Claims 50 and 52 are rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and Langberg et al. (US 5,852,630) and further in view of Shimokawa et al. (US 4,298,933).

(1) with regard to claim 50 :

Albert et al. further discloses logically AND, as each Condition's Boolean value is computed, the Boolean values resulting from evaluating the Condition(s) (Fig. 12 and paragraph [0232] teaches that the result of each condition are logically ANDed, i.e. all have to be YES in order to a MATCH, either one is NO will result in a NO MATCH). And since Albert et al. discloses the result of each condition comparison only requires 1 bit (to represent 1 or 0 representing YES or NO), a 1-bit device is used for performing this method.

Albert et al. however does not expressly disclose utilizing a Condition Accumulator to perform such method.

Shimokawa et al. teaches the use of accumulator for storing intermediate arithmetic and logic results (Bit Accumulator, 32 on Fig. 5). It would have been desirable to use an accumulator for storing intermediate logic results because it would eliminate the need for writing the result of each calculation to a separate memory device, thus improve the efficiency of the operation. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use accumulator for storing logical operation results in the system of Albert et al.

(2) with regard to claim 52:

Albert et al. further discloses logically OR, as each Condition Set's Boolean value is computed, the Boolean values resulting from determining if the Condition Set is met (Fig. 12 and paragraph [0232] teaches that the final result is either a MATCH or NO MATCH, this indicates that MATCH or NO MATCH is logically ORed). And since Albert et al. discloses the final result only requires 1 bit (to represent 1 and 0 representing a MATCH or NO MATCH), a 1-bit device is used for performing this method.

Albert et al. however does not expressly disclose utilizing a Condition Accumulator to perform that method.

Shimokawa et al. teaches the use of accumulator for storing intermediate arithmetic and logic results (Bit Accumulator, 32 on Fig. 5). It would have been desirable to use an accumulator for storing intermediate logic results because it would eliminate the need for writing the result of each calculation to a separate memory

Art Unit: 2619

device, thus improve the efficiency of the operation. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to use accumulator for storing logical operation results in the system of Albert et al.

16. Claim 55 is rejected under 35 U.S.C. 103(a) as being unpatentable over Albert et al. (US 2002/0141401 A1) in view of Relan (US 2003/0174703) and Langberg et al. (US 5,852,630). and further in view of Egbert et al. (US 6,115,387).

(1) with regard to claim 55:

Albert et al. does not disclose the Condition Indirection Table is stored within a Content Addressable Memory (CAM).

Egbert et al. teaches using a content addressable memory to store a table that stores pointers (161, on Fig. 14).

It would have been desirable to use a content addressable memory to store tables because it would reduce system latency as table look-ups with content address memory is very fast and efficient. Therefore, it would have been obvious to one of ordinary skill in the art at the time of the invention to include the method as taught by Egbert et al. in to the system of Albert et al.

Response to Arguments

17. Applicant's arguments with respect to claims 1 – 12, 14 – 30, 32 – 52 and 54 - 60 have been considered but are moot in view of the new ground(s) of rejection.

With regard to the 112 2nd rejections of claims 5, 6, 45 and 46, Applicant in response argues that there is only one incidence of "Rules Table" in claims 5 and 45, and their parent claims. The Examiner respectfully disagrees. Each of their respective parent claim also introduce a "Rules Table" and it is not clear whether the two Rules Tables are referring to the same Rules Table or otherwise.

Conclusion

18. Applicant's amendment necessitated the new ground(s) of rejection presented in this Office action. Accordingly, **THIS ACTION IS MADE FINAL**. See MPEP § 706.07(a). Applicant is reminded of the extension of time policy as set forth in 37 CFR 1.136(a).

A shortened statutory period for reply to this final action is set to expire **THREE MONTHS** from the mailing date of this action. In the event a first reply is filed within **TWO MONTHS** of the mailing date of this final action and the advisory action is not mailed until after the end of the **THREE-MONTH** shortened statutory period, then the shortened statutory period will expire on the date the advisory action is mailed, and any extension fee pursuant to 37 CFR 1.136(a) will be calculated from the mailing date of

Art Unit: 2619

the advisory action. In no event, however, will the statutory period for reply expire later than SIX MONTHS from the date of this final action.

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Bo Hui A. Zhu whose telephone number is (571)270-1086. The examiner can normally be reached on Mon-Thur 10am-6pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Hassan Kizou can be reached on (571)272-3088. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

BZ
Examiner
October 16, 2007



HASSAN KIZOU
SUPERVISORY PATENT EXAMINER
TECHNOLOGY CENTER 2600